**PuzzLink – Details Design**

**Partners:** Adi Vered (318243755), Shiran Reich (318910650)

**General Project Description**

The goal of the **PuzzLink** application is to provide a social platform that allows users to communicate and collaborate regardless of physical distance. The system enables the creation of private rooms, real-time communication through a shared board or collaborative creative activities (e.g., uploading and designing images), and chat. Additionally, it allows users to draw, erase, and hold real-time conversations.

**Project Objectives**

* Facilitate social activities without dependency on physical proximity.
* Enable image sharing between users.
* Create an interactive and enjoyable problem-solving experience.

**Target Audience**

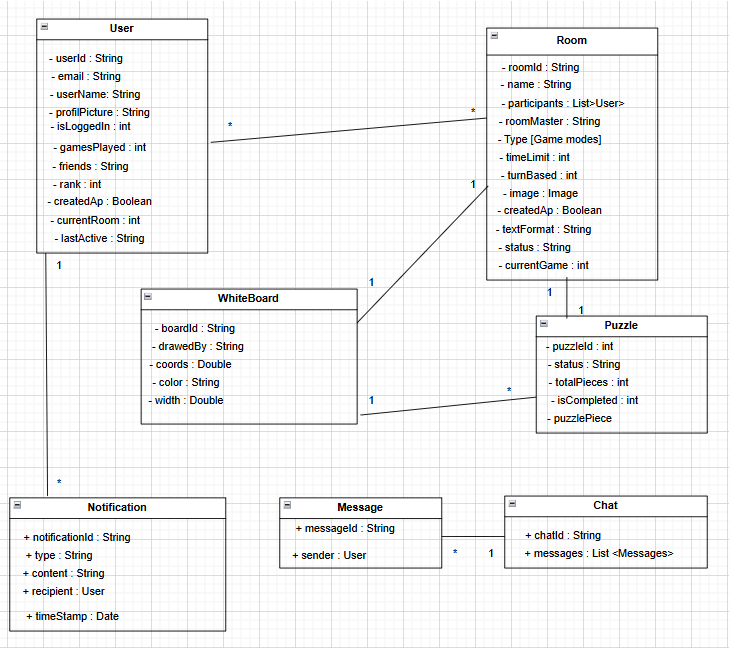
* Friends/couples looking for shared activities.
* Educational communities interested in using the app for social purposes.
* Professional institutions in education, diagnostics, and psychology.

**Functional Requirements**

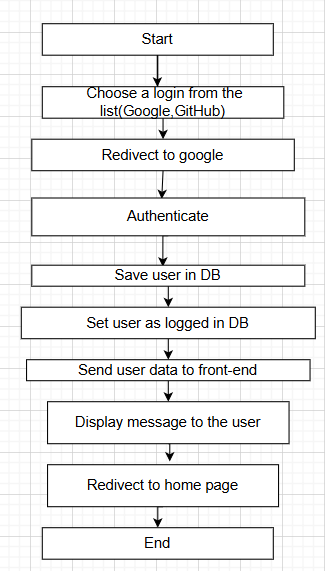
1. *User Authentication*
   * Users can sign up for an account using their email, username, and password.
   * Users can log into their account with credentials or via third-party authentication (e.g., Google, GitHub).
   * Users will be tokenized and remembered with cookies.
2. *Private Rooms*
   * Users can create and join rooms.
   * Users can navigate to a shared web page within the room.
   * **Note**: The core requirement is that multiple users can write on the shared board in real-time, which is the platform's primary function.
3. *Room Type*
   * Users can choose between a room with an image or a blank board.
4. *Image Upload*
   * Users can upload an image or generate one based on a text prompt.
   * After uploading, users can engage in drawing activities on the board.
5. *Room Options*
   * Users can draw or erase on the board.
   * Users can adjust pen width and color.
6. *Private Room Chat*
   * Users can access a chat corresponding to their room.
   * Users can send and receive text messages in the chat.
   * Real-time updates will be provided for new messages.
7. *Profile Tab*
   * Users can view and edit their profile details, including name, contact information, and recent drawings.
   * Users can upload a profile picture.
   * Users can view their activity history within the app (e.g., previous rooms, collaborators).
8. *Notifications*
   * Users will receive notifications for new chat messages.
   * Users will receive notifications for new friend requests.

**Data Structure - Diagram**

* **Class Diagram**



* **Activity Diagram:** Sequence of actions in the user login process for the application



**System Architecture and Technologies:**

**Server-Side Architecture:**

* **Application Server**: Node.js
* **Database**: MongoDB

**Client-Side Architecture:**

1. **React**
2. **State Management and Storage**: Redux
3. **Styling**: CSS3
4. **Animations**: GSAP

**Communication Between Server and Client:**

* **Security Mechanism**: Middleware for traffic management
* **Session Management System**: Express
* **Routing and Data Access System**: API-like structure
* **Web Sockets**: For real-time updates (e.g., tracking puzzle progress).

**Security:**

* User authentication using JWT.
* Third-party authentication for added security (e.g., Google).